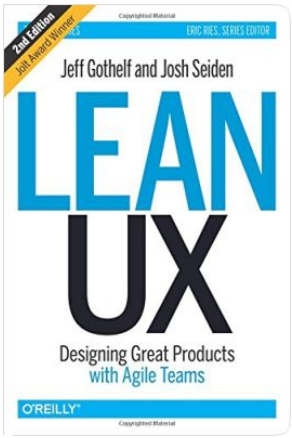

Lean UX

Designing Great Products with Agile Teams

Jeff Gothelf and Josh Seiden | O'Reilly © 2016

In this summary, you will learn: 1) What strategies to use to implement a Lean approach to designing a product for an end user and 2) How to integrate Lean and Agile development.

Take-Aways

- Lean UX follows the principles of Lean development.
- Lean UX concerns outcomes, not deliverables.
- Eliminate “Big Design Up Front” and follow the iterative, ongoing continuous improvement principles of Lean development.
- Lean UX relies on cross-divisional collaboration that breaks down silos.
- Work as a team to determine your “hypotheses” – the designs you believe will most likely deliver the desired outcome.
- Know your customers. Have your team develop “proto-personas” to represent them.
- Prioritize your hypotheses – and the features and assumptions you want to test – according to how your personas achieve their goals through using that feature.
- Create “minimum viable products” (MVPs) or “prototypes” to test your hypotheses.
- Prove or disprove each product as quickly and cheaply as possible.
- Cull designs that customers reject and dismiss team members who won’t collaborate.

Recommendation

Lean UX combines the Lean Principles of “rapid experimentation” and “evidence-based decision making,” with systems thinking and an agile approach to development. As in Lean development, Lean UX emphasizes supporting the user experience, achieving a “minimum viable product” (MVP), and sharing the MVP while collecting feedback and making rapid revisions. Lean

UX adds a critical component to the genre. Authors Jeff Gothelf and Josh Seiden, who are design experts, present their information clearly and cleanly, though they do seem to figure that readers already have a grounding in Lean and Agile development. Thus, *getAbstract* recommends this fluff-free and practical guide to designers, developers, program managers and project managers.

Summary

Why Lean UX?

Today's companies send out a continual flow of digital products and updates. Amazon makes updates, on average, every 11.6 seconds. Your ability to make constant improvements in response to ongoing user feedback creates a better customer experience and a stronger business.

> *"Lean UX is not a set of rules. Instead, it's an approach that you adopt."*

In contrast, those who follow the old "waterfall" approaches to development can't compete with this faster process. Detailed and drawn-out design procedures can't keep up with the speed at which makers now must adapt their products, services and features. The Lean UX approach offers processes you can adapt to your culture, even as it changes your culture.

"The Foundations of Lean UX"

Lean UX relies on "design thinking" that emphasizes collaboration. It calls for a cross-disciplinary, open and inclusive mind-set that prioritizes action and feedback over planning and documentation. Lean UX designers and developers expect their early releases to miss the mark, but after release they iterate the product constantly toward success.

Lean UX Principles

These principles guide the Lean UX culture, its teams and its processes:

- **Lean Culture** – Start by labeling everything as opinion, not fact. Test all opinions with customers. Base your decisions about features and tools on evidence. Discard or park any feature that fails consumer testing. Encourage risk taking.
- **Lean Teams** – Lean UX teams have the autonomy to determine solutions to actual business issues. Members collaborate throughout the product's life cycle. The team should not exceed 10 members.
- **Lean Process** – Work on small parts of the design so you don't go too far before realizing you need to change course. Watch customers to learn how they interact with your design. This means leaving your building.

The Lean UX Process

Concern yourself with outcomes, not deliverables. Instead of defining a feature, building it and hoping for the best, go straight to the results you want. For example, instead of saying "We need an app," say, "We want to increase revenue through mobile." Don't identify product "requirements." Call them "assumptions" so team members understand that whatever they create will change. You might end up developing an app, but you might also discover a better path to achieving the outcome you desire.

> *"Collaborative design is still a designer-led activity. It's the designer's responsibility to not only call collaborative design meetings but to facilitate them, as well."*

Determine your hypotheses based on reaching your desired outcome, moving fast and avoiding using up time creating designs that don't work. Figure out how to prove or disprove each assumption quickly and cheaply.

Find and state assumptions by bringing multidisciplinary teams and subject matter experts together. State the problem clearly, including goals, gaps and facets that need improvement. Have group members study the problem individually. For new products, define the current marketplace, the gaps you seek to fill and your strategic vision for how you'll fill it.

> *"Focus on maximizing two factors: increasing collaboration between client and agency, and working to change the focus from outputs to outcomes."*

Start with a statement of the problem. Have the team discuss the range of assumptions inherent in the problem statement. Turn the assumptions into statements of hypothesis in which you declare what you believe to be true. Describe how you'll test the truth of your hypotheses according to quantitative and qualitative data about the outcomes.

> *"Besides negotiating the complexity of certain features, cross-functional participation allows designers and developers to create effective backlog prioritization."*

Gather team assumptions about your typical users. These "proto-personas" should include demographics – age, gender, marital status, income, interests, and the like. Revise your personas as you learn about your clientele. Your users don't care about functionality – they care about meeting their goals. Learn about your customers' goals and the barriers in their way.

Test Your Hypotheses

Test your hypotheses by stating the outcomes you wish to achieve. Prioritize your hypotheses and the features and the assumptions you want to test according to the extent to which your personas achieve their goals by using that feature. Assess risk – how harmful would it be to act on this assumption if it turns out to be wrong?

> *"MVPs help us test our assumptions – will this tactic achieve the desired outcome? – while minimizing the work we put into unproven ideas."*

Use the following process to find your best ideas, create a MVP and conduct your research:

Collaborate

Collaborate with designers and others to create the products, merchandising and support that add up to the overall user experience. Manage the process, but involve others early on. More diverse input into design ideas leads to better results. Collaborating brings your team together around the eventual design and increases their design competence. Sketch out ideas and build rough "wireframes," leaving room for rapid change and iteration. As your team grows more comfortable – normally after several informal sessions – plan a formal "Design Studio" meeting. Set aside several hours for planning in groups of five to eight team members.

> *"In Agile, the entire team must participate in all activities – stand-ups, retrospectives, IPMs, brainstorming sessions – they all require everyone's attendance to be successful."*

Look first at the problem and barriers. Ask each person to brainstorm and roughly sketch ideas for group discussion. Each person should take a few minutes to share a sketch and describe the problem it addresses. Put people in pairs, combining those with similar ideas, and have them refine the ideas further. Give them 10 minutes and have the pairs present again. Select the best ideas together. Allot three-quarters of an hour for discussion and voting.

> *"Collaboration yields better results than hero-based design (the practice of calling a designer or design team to drop in, come up with something beautiful and take off)."*

As a group, select an idea to pursue. Create a "Design System" document, including brand standards. This sets out rules and provides examples for various teams to follow while working on different designs. It avoids duplication of effort, as it accelerates the process through the reuse of expert designs and approved foundational elements. Bring your disparate team together to build your Design System. Evolve it continuously. Build a website repository for sharing. Assign someone to keep it current and "accessible."

Create "Minimum Viable Products" (MVP)

To test your ideas and hypotheses, build "prototypes" of your MVP to get your designs out quickly and to learn what people want. For example, you might hypothesize that you can increase market share by producing a monthly newsletter. That requires a lot of work, so test the demand. Your MVP might consist of a newsletter "sign-up" with an attractive sketch of a cover design.

> *"This prioritized list of risks defines the work that the team must continue to validate throughout the project life cycle."*

Don't bother with the niceties; a sketch or a picture will do. You can make a prototype using paper and tape, or you can do it online using simple wireframes. More sophisticated prototypes will closely mimic the finished product, especially those using code with real or realistic data. Such prototypes offer the truest simulations, but take the most time and resources. Gauge your effort to the degree of proof you need and how valuable you think the feature or product will be.

Make Prototypes

Share your prototypes widely. Observe as insiders interact with them and ask users questions. After you've shared an MVP internally and adjusted it, take it out to customers. Know what you want to learn from testing your MVP. It should generate sufficient concrete feedback – usually by asking people to sign up or opt in to something. Unless you have a high-traffic site, you might use a campaign to attract people. To get your MVPs in front of a lot of people, consider using existing vehicles, such as sites like Kickstarter, Facebook Groups or eBay storefronts. Depending on the number of sign-ups or opt-ins, either terminate the project or test other assumptions as you progress into more design and development.

Gather Feedback and Research

With your MVP fully developed, validate it with market research. Conduct your own research; don't outsource it. Decide whom you need to speak with, develop an interview worksheet, talk with customers and show them the MVP. Ask questions, take notes and ask for referrals to other potential interviewees. Compare notes; look for themes and patterns. Change your MVP or prototype before the next round of interviews. Repeat the process from MVP to product. Analyze and interpret the data as a team.

> *"Lean UX radically shifts the way we frame our work by introducing back the strategic context for our feature and design choices and, more important, how we – the entire team, not just the design department – define success."*

Look for group themes and patterns in the data. Set aside interesting "outliers" for further review. It may take weeks of sessions to see patterns and test outliers. Conduct customer research after your product launch. Attitudes and preferences change, but research alerts you to these changes almost in real time. Regularly collect information from your customer service staff. Learn what customers search for on your site, what they find, what they only look at, what they use or interact with, and where they exit. Check your assumptions and hypotheses by releasing slightly different MVPs to different user groups. Repeat often to iterate faster and better.

“Integrating Lean UX”

As you make the transition to Agile, short design cycles should precede short development cycles. Integrate your design and development teams for “dual track agile”: A “discovery” team focuses on testing hypotheses and assumptions and talking to users. The development team builds the ideas that survive discovery. Some delivery teams participate in discovery and vice versa.

> *“Design only what you need. Deliver it quickly. Create enough customer contact to get meaningful feedback fast.”*

After a product or a new feature launches, collect data for discovery. Integrate Lean UX within agile “scrums.” Start with a “design sprint.” Name the hypothesis. Describe how you will test it, who’s responsible and the estimated effort. Do weekly market research sessions with customers to ensure feedback to developers and sprints. Include designers in prioritizing work for the sprints. Communicate your plans and programs to stakeholders.

“Supporting Lean UX”

Adjust your operations to optimize Lean UX by following 15 rules:

1. **You can’t predict your product** – Test your ideas and assumptions. Stay open to experimentation and iteration. Drop products and ideas if experimentation doesn’t support them.
2. **Focus on outcomes, not deliverables** – Leaders name outcomes, but teams decide the features that get them there.
3. **Break down internal silos** – Enable cross-disciplinary teams to collaborate according to members’ knowledge, not their job titles.
4. **Have designers work with the whole team** – Developers, product managers and talent from each discipline must collaborate.
5. **Assign smaller teams to smaller problems** – Align their work to a single result.
6. **Bring your teams together physically** – Find an open space with walls for posting, and table space and breakout rooms for brainstorming.
7. **Use collaborative online tools** – Plan online meetings so everyone can attend, regardless of time zones.
8. **Remove heroes** – Lean UX has no stars; its teams act on customer evidence.
9. **Don’t fall for “Big Design Up Front”** – Lean design follows the iterative, MVP principles of Lean development.
10. **Don’t worry about appearances** – Design ugly, fast wireframes or prototypes of paper and tape. Test ideas before making the design beautiful.
11. **Commit to continuous improvement in design** – Revisit the user experience to finish features left in your “UX debt.”

12. **Make agreements with outsourcers** – Develop "time and materials" contracts to align willing outsourcers to the Lean UX pace.
13. **Start with notes, sketches, experiments and conversations** – Once you've tested hypotheses and you know what you'll build, follow the formal documentation process to create references for future teams.
14. **Consider the perspective of others** – Aim for quick wins to show the process's value.
15. **Communicate** – Inform stakeholders of your progress and plans.

About the Authors

Consultants and speakers **Jeff Gothelf** and **Josh Seiden** have more than four decades of combined digital design experience.